

## PEMBANGUNAN APLIKASI PEMODELAN DIAGRAM KELAS UNTUK PEMBELAJARAN UNIFIED MODELING LANGUAGE

*Egia Rosi Subhiyako<sup>1</sup>, Yani Parti Astuti<sup>2</sup>*

<sup>1,2</sup>Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro  
e-mail: <sup>1</sup>[egia@dsn.dinus.ac.id](mailto:egia@dsn.dinus.ac.id), <sup>2</sup>[yanipartiastuti@dsn.dinus.ac.id](mailto:yanipartiastuti@dsn.dinus.ac.id)

### ABSTRAK

*Pengembangan perangkat lunak yang dilakukan oleh seorang pengembang biasanya memperhatikan tahapan pengembangan. Dalam fase analisis, yakni memodelkan masalah ke dalam bentuk diagram seperti DFD untuk analisis terstruktur dan UML untuk analisis berorientasi objek. Masalah yang sering ditemui didunia pendidikan yakni dalam pemodelan UML, alat pemodelan yang sudah ada memiliki kompleksitas yang tinggi dan mempunyai lisensi sehingga mahasiswa tidak bisa secara langsung menggunakan alat pemodelan tersebut. Dalam penelitian ini dikembangkan sebuah aplikasi pemodelan UML khususnya diagram kelas yang berbasis web, open access, memiliki tampilan sederhana, dan memiliki fitur-fitur dasar untuk memodelkan diagram kelas. Diagram kelas mempresentasikan atribut dan method yang digunakan untuk menggambarkan sebuah sistem. Aplikasi ini di kembangkan dengan menggunakan metode prototyping, aplikasi dikembangkan sesuai spesifikasi awal, setelah itu disembarkannya skala kepuasan untuk mengetahui aplikasi bekerja dengan baik atau tidak. Lalu dilakukan evaluasi untuk menentukan aplikasi siap digunakan, berdasarkan evaluasi dapat disimpulkan bahwa sebagian besar pengguna menyetujui kinerja aplikasi baik, dengan prosentase Sangat Setuju dan Setuju 88.01% responden. Kemudahan pengguna dengan prosentase Sangat Setuju dan Setuju 83.86% responden. Kepuasan pengguna Sangat Setuju dan Setuju 87.09% responden. Dengan hasil tersebut dapat disimpulkan mayoritas responden setuju bahwa kinerja aplikasi baik dan mudah digunakan serta pengguna puas terhadap aplikasi yang dibangun.*

**Kata Kunci:** UML, kelas, diagram, prototyping

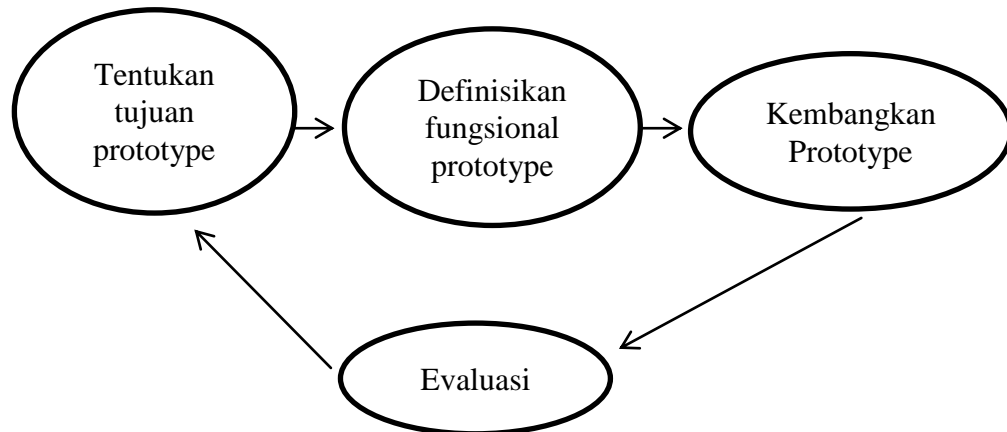
### 1. PENDAHULUAN

Perkembangan dunia rekayasa perangkat lunak dewasa ini semakin meningkat dengan munculnya berbagai macam *start up*. Pada dasarnya, pengembangan perangkat lunak dapat dibagi menjadi dua dunia, yaitu dunia industri dan dunia pendidikan. Dalam dunia pendidikan pengembangan perangkat lunak ditujukan untuk dunia pendidikan. Dalam dunia pendidikan mahasiswa biasanya mengambil peran sebagai pengembang perangkat lunak pemula. Pada dasarnya untuk menjadi pengembang perangkat lunak yang kompeten siswa harus memperoleh pengetahuan dan keterampilan yang mencakup semua tahapan pengembangan[1]. Pada tahapan pengembangan salah satunya adalah tahapan analisis, pada tahapan ini siswa harus bisa melakukan tahapan pemodelan berupa pemodelan kebutuhan perangkat lunak baik menggunakan pemodelan terstruktur maupun berorientasi objek.

Pemodelan berorientasi objek menggunakan bahasa pemodelan UML (*Unified Modeling Language*). Pada saat ini, alat pemodelan yang digunakan untuk melakukan pemodelan kebutuhan rekayasa perangkat lunak dengan UML telah banyak tersedia, namun dikembangkan untuk tujuan industri dan tidak diberikan sebuah teori yang dibutuhkan untuk mengajar contohnya StartUML, ArgoUML, dan Marama[1]. Adapun alat dasar pemodelan UML yang berfokus pada peningkatan pembelajaran, seperti *Enterprise Architect* tetapi alat pemodelan ini berbayar sehingga menyulitkan siswa untuk memakainya. Penelitian yang membahas penilaian kerjasama tim pemula dilakukan dalam [2].

Dalam penelitian yang dilakukan oleh [3] biasanya Case Tool yang digunakan berbasis *desktop* dan membutuhkan penginstalan yang cukup panjang untuk setiap komputer yang akan dipergunakan, sehingga terbatas pada sistem operasi tertentu, dan media yang terbatas pula dalam mengakses aplikasi tersebut, sehingga membutuhkan spesifikasi komputer yang tinggi untuk menjalankan aplikasi. Penelitian ini juga menjelaskan tentang pemodelan UML berbasis web yang menggunakan metode *Scrum Model* tanpa membutuhkan penginstalan dan spesifikasi yang besar untuk menggunakan aplikasi, *project UML* disimpan secara *online* dan pengguna dapat membuka kembali. Penelitian lain dalam [4] membangun sebuah aplikasi yang bernama RMTTool, yang dikembangkan berbasis *web* dengan JSP sebagai bahasa pemrograman dan MySQL sebagai basis datanya. Penelitian lainnya pada [5] sebagian siswa kesulitan dalam merancang, Web UML yang mengembangkan LogViz untuk memantau aktivitas dalam merancang menunjukan dengan memvisualisasikan aktivitas siswa untuk membuat desain. Sedangkan dalam penelitian lain pengembangan alat pemodelan UML (*Unified Modeling Language*) untuk pendidikan yang berbasis web dan *open access* berupa aplikasi yang bernama Light Case. Penelitian yang sudah dipaparkan sebelumnya berhubungan dengan topik yang akan diangkat yaitu Pengembangan Alat Pemodelan UML khususnya diagram kelas. Alat pemodelan yang dibangun ini berbasis web dan *open access*.

## 2. METODE PENELITIAN



Gambar 1. Metode Prototyping.

Metode yang dipakai pada pengembangan aplikasi ini adalah metode *prototyping*. Tahapan pengembangan berdasarkan gambar 1 adalah sebagai berikut:

1. Menentukan tujuan prototype, dalam pengembangan aplikasi ini tujuannya adalah untuk memudahkan mahasiswa melakukan pemodelan rekayasa perangkat lunak.
2. Mengembangkan aplikasi yang disesuaikan dengan kebutuhan dalam melakukan pemodelan rekayasa perangkat lunak.
3. Melakukan pengujian perangkat lunak kepada beberapa mahasiswa sebagai sampel untuk menentukan hasil pengujian.
4. Mengetahui hasil dengan memberikan skala kepuasan pengguna terhadap perangkat yang telah dikembangkan. Jawaban yang telah diberikan oleh mahasiswa sebagai hasil koreksi yang berguna untuk perbaikan sistem agar sesuai dengan kebutuhan pengguna.

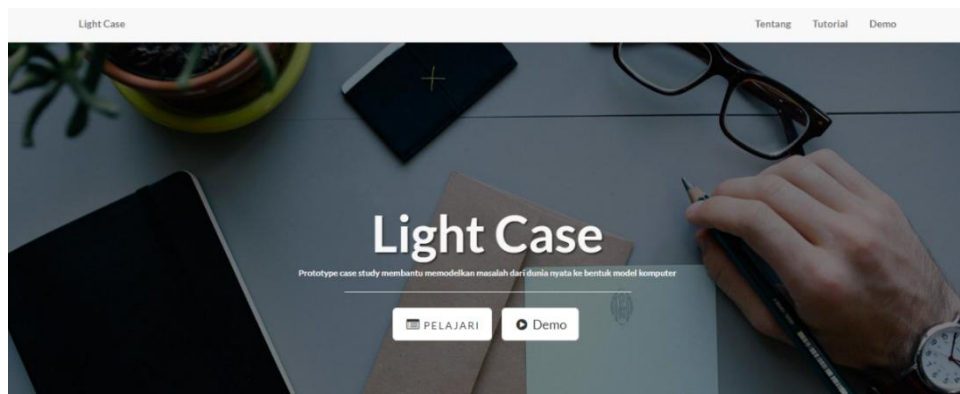
## 3. HASIL DAN PEMBAHASAN

Bagian ini meliputi hasil implementasi dan pengujian terhadap aplikasi yang dibangun.

### 3.1 Implementasi

Sistem yang dikembangkan merupakan aplikasi yang berbasis web, dapat diakses dengan cepat dan ringan. Pada aplikasi ini didalamnya terdapat studi kasus yang akan diselesaikan oleh pengguna, setelah pengguna memahami studi kasus yang diberikan, pengguna akan menyelesaikannya dengan diagram kelas. Pengguna dapat menggunakan fitur *hint* untuk membantu merancang, dan hasil diagram yang dibuat dapat di simpan atau di *export* ke dalam format image.

Aplikasi yang dikembangkan kami beri nama *Light Case*. Gambar 2 menunjukkan tampilan utama aplikasi yang dibangun.

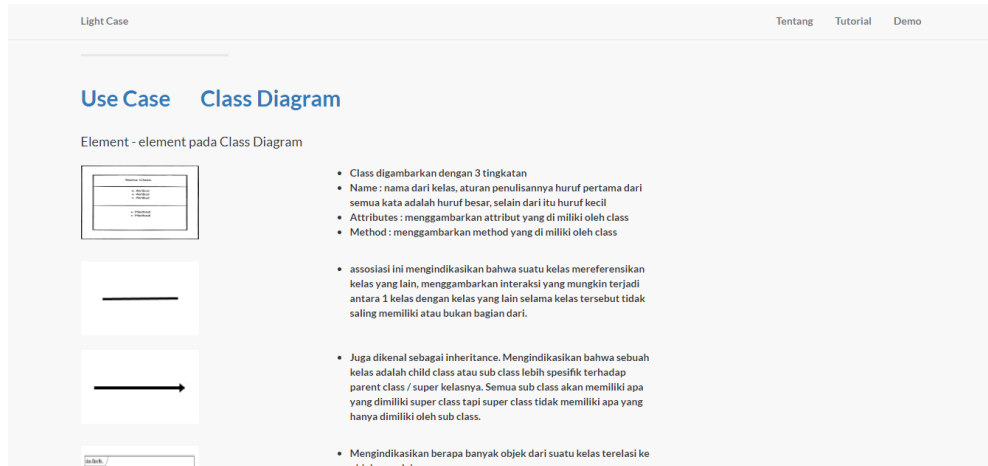


Gambar 2. Halaman Utama Aplikasi.

Gambar 3 menunjukkan halaman tutorial penggunaan diagram kelas.

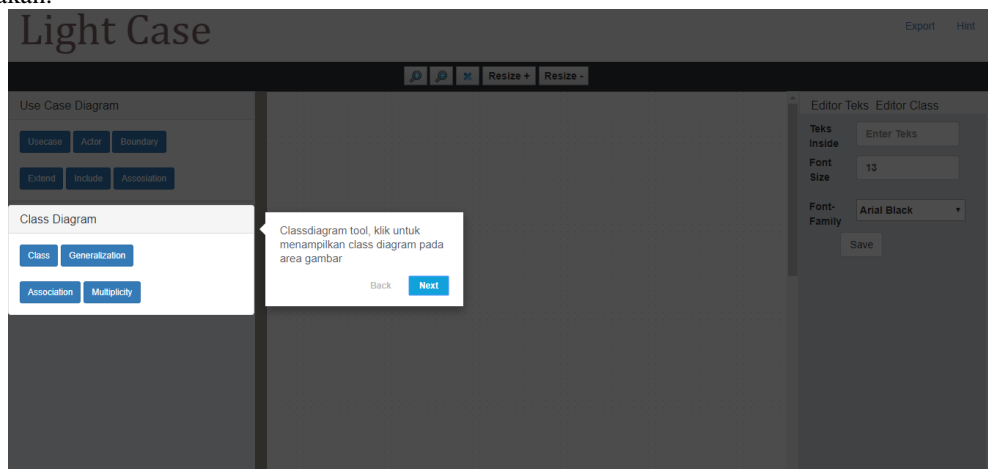
No	Pernyataan	Nilai								Jumlah Point	Point Rata-rata	Presentase
		SS	S	TS	STS	SS*4	S*3	TS*2	STS*1			
<b>Kinerja Sistem</b>												
1	Sistem mudah digunakan	10	20	1	0	40	60	2	0	102	3.290	82.25
2	Sistem membantu memperkenalkan pengetahuan diagram kelas	10	18	3	0	40	54	6	0	100	3.225	80.625
3	Sistem memadai untuk menyelesaikan studi kasus	15	12	4	0	60	36	8	0	104	3.354	83.85
4	Sistem memiliki kinerja yang baik	13	12	4	1	52	36	8	2	98	3.161	79.032
5	Sistem dapat bekerja dengan baik	10	19	1	1	40	57	2	1	100	3.225	80.625
6	Sistem responsif	8	16	6	1	32	48	12	1	93	3	75
7	Sistem mampu membantu mengenalkan tentang diagram kelas	9	19	3	0	36	57	6	0	99	3.193	79.825
<b>Kemudahan Pengguna</b>												
8	Antarmuka sistem menarik	10	15	6	0	40	45	12	0	98	3.161	79.025
9	Antarmuka mudah dipahami dan digunakan	11	18	2	0	44	54	4	0	102	3.290	82.25
10	Sistem membantu pengguna memahami diagram kelas	16	10	5	0	64	30	20	0	114	3.677	91.925
11	Sistem membantu pengguna membuat diagram kelas	12	17	1	1	48	51	2	1	102	3.290	82.25
12	Sistem membantu memahami definisi diagram kelas	8	18	5	0	32	54	70	0	96	3.096	77.4
13	Sistem membantu pengguna mengubah dan memodifikasi diagram kelas	14	13	3	1	56	39	6	1	102	3.290	82.25
<b>Kepuasan Pengguna</b>												
14.	Saya senang dengan	9	19	3	0	36	57	6	0	99	3.193	79.825

	antarmuka sistem.											
15	Saya dengan fungsional sistem.	9	19	3	1	32	57	6	1	96	3.096	77.4
16	Saya senang dengan tampilan diagram kelas yang saya buat.	7	16	6	1	28	48	12	1	89	2.870	71.75



Gambar 3. Halaman Tutorial Diagram Kelas.

Gambar 4 menunjukkan menu yang dapat digunakan untuk membuat diagram kelas di dalam area yang sudah disediakan.



Gambar 4. Halaman Menu Diagram Kelas.

### 3.2 Pengujian

Evaluasi yang digunakan adalah dengan melakukan pengujian *black box* dan kuesioner *user acceptance*. Teknik pengujian seperti yang dipaparkan dalam [6] juga menjadi acuan penelitian ini. Sampel diambil sebanyak 31 responden dari mahasiswa Teknik Informatika Universitas Dian Nuswantoro yang telah mengambil mata kuliah Rekayasa Perangkat Lunak. Responden diberikan 16 pertanyaan dengan 4 parameter penilaian meliputi, “Sangat Setuju” dengan bobot nilai 4, “Setuju” dengan bobot nilai 3, “Tidak Setuju” dengan bobot nilai 2, “Sangat Tidak Setuju” dengan bobot nilai 1. Berikut hasil kuisisioner:

Tabel 1. Hasil Kuisisioner Pengujian Aplikasi.

Keterangan :

SS = Sangat Setuju

S = Setuju

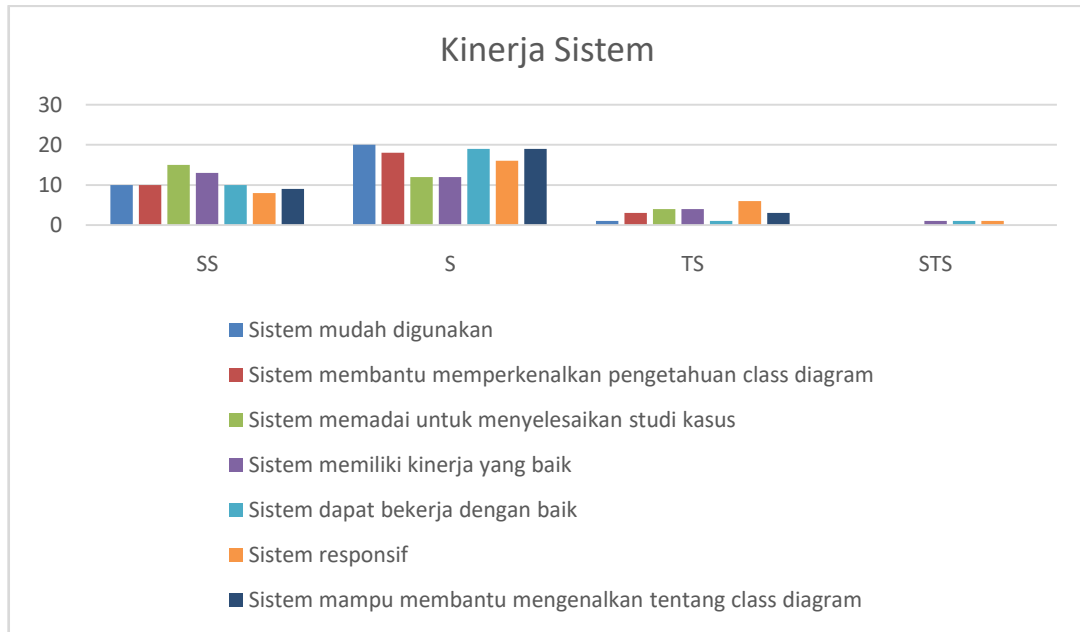
TS = Tidak Setuju

STS = Sangat Tidak Setuju

$$\text{Jumlah point} = \text{SS} * 4 + \text{S} * 3 + \text{TS} * 2 + \text{STS} * 1$$

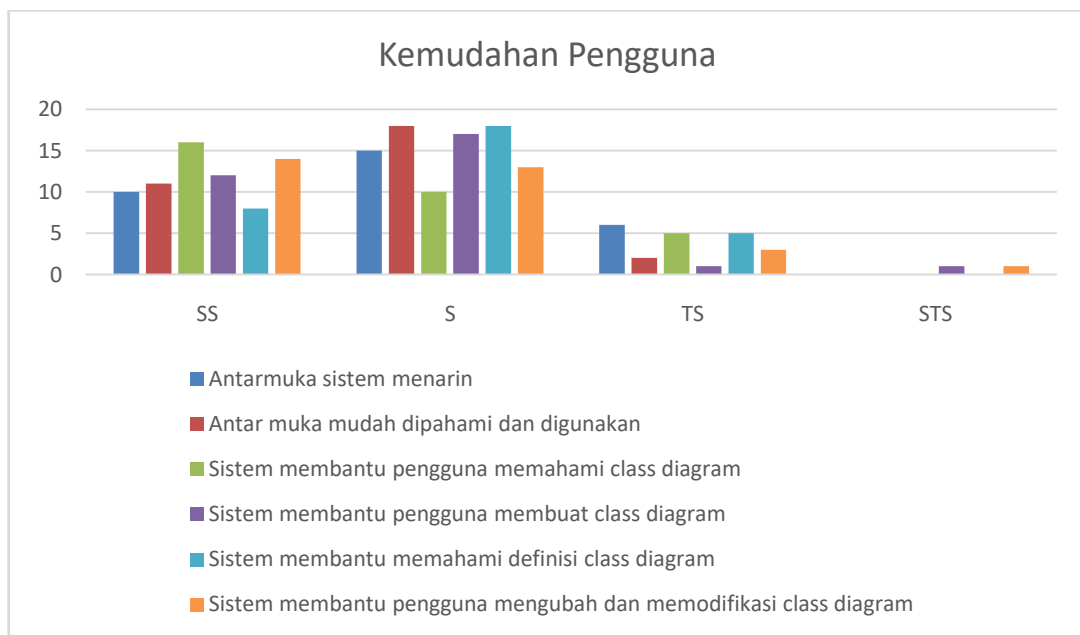
$$\text{Point rata-rata} = \text{Jumlah point} / \text{Jumlah Responder}$$

$$\text{Nilai dalam \%} = \text{Point rata-rata} / 4 (\text{Jumlah Parameter}) * 100$$



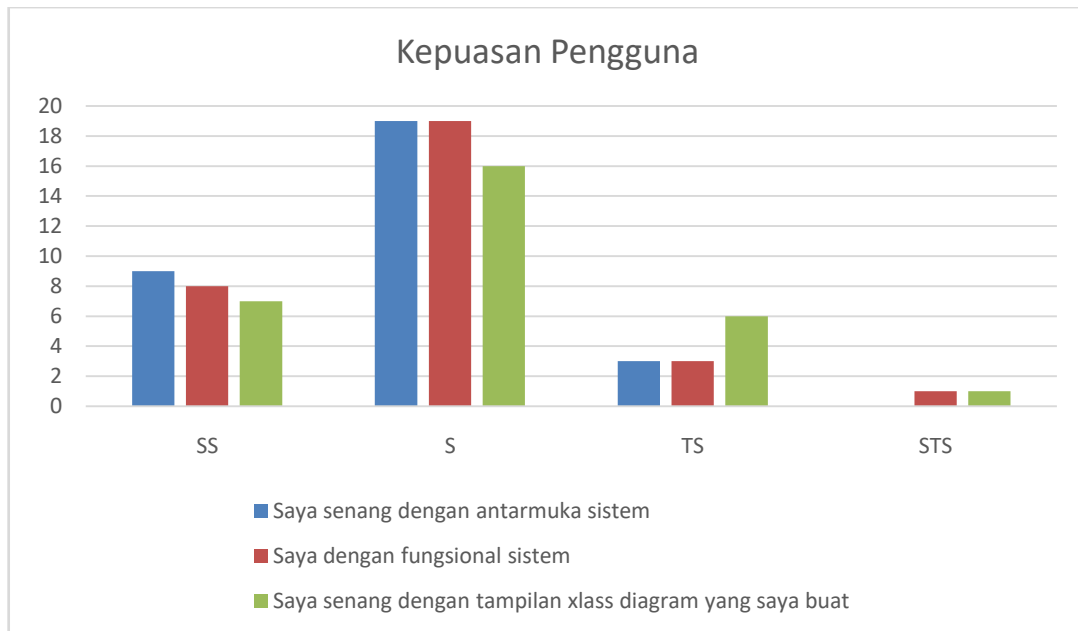
Gambar 5. Grafik Kinerja Sistem.

Grafik 5 diatas menunjukkan sebagian besar pengguna menyetujui kinerja sistem Light Case baik, dengan prosentase “Sangat Setuju” dan “Setuju” 88.01% responden. Sedangkan presentase “Tidak Setuju” dan “Sangat Tidak Setuju” 11.52% responden.



Gambar 6. Grafik Hasil Kemudahan Pengguna.

Grafik 6 menunjukkan sebagian besar pengguna menyetujui kemudahan pengguna Light Case baik, dengan prosentase “Sangat Setuju” dan “Setuju” 83.86% responden. Sedangkan presentase “Tidak Setuju” dan “Sangat Tidak Setuju” 11.82% responden.



Gambar 7. Grafik Hasil Kepuasan Pengguna.

Grafik 7 diatas menunjukkan sebagian besar pengguna menyetujui kepuasan pengguna Light Case baik, dengan prosentase “Sangat Setuju” dan “Setuju” 87,09% responden. Sedangkan presentase “Tidak Setuju” dan “Sangat Tidak Setuju” 12,87% responden.

#### 4. KESIMPULAN

Telah dikembangkan prototype aplikasi pemodelan UML khususnya diagram kelas yang diberi nama Light Case. Hasil evaluasi yang dilakukan terhadap kinerja sistem, kemudahan penggunaan, dan kepuasan pengguna terhadap Light Case, mendapatkan hasil sebagai berikut; Sebagian besar pengguna (88,01%) menilai kinerja sistem Light Case sudah cukup baik dan menunjukkan grafik yang positif, mayoritas pengguna (83,86%) menilai kemudahan penggunaan dalam Light Case sudah baik, dan mayoritas pengguna puas dengan aplikasi Light Case dengan prosentase 87,09%. Penelitian selanjutnya adalah menambahkan diagram-diagram lain dalam UML seperti diagram aktivitas, diagram sequence, dan diagram state.

#### UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada LPPM Universitas Dian Nuswantoro Semarang yang telah memberi “dukungan financial” terhadap penelitian ini, reviewer, editor dan semua pihak yang telah membantu dalam penyelesaian paper ini.

#### DAFTAR PUSTAKA

- [1] E. R. Subhiyakto, M. Kamalrudin, S. Sidek, and S. S. S. Ahmad, “Customization of Requirements Modelling Tools for Software Engineering Education,” *Int. Symp. Res. Innov. Sustain. 2014*, vol. 26, no. 4, pp. 1581–1584, 2014.
- [2] D. Wahyu, Utomo, and E. R. Subhiyakto, “Assessing Novice Teams in Collaborative Software Engineering Education,” in *Seminar Nasional Aplikasi Teknologi Informasi (SNATi)*, 2016, pp. 23–28.
- [3] M. Subekti, D. Indrawan, and G. Putra, “Perancangan Case Tools untuk Diagram Use Case, Activity, dan Class untuk Permodelan UML Berbasis Web Menggunakan HTML5 dan PHP,” *ComTech*, vol. 5, no. 2, pp. 625–635, 2014.
- [4] E. R. Subhiyakto and D. W. Utomo, “RMTTool; Sebuah Aplikasi Pemodelan Persyaratan Perangkat Lunak menggunakan UML,” *JNTETI*, vol. 6, no. 3, pp. 268–274, 2017.
- [5] D. R. Stikkolorum, T. Ho-quang, and Chaudron M.R.V, “Revealing Students ’ UML Class Diagram Modelling Strategies with WebUML and LogViz,” in *41st Euromicro Conference on Software Engineering and Advanced Applications Revealing*, 2015.
- [6] E. R. Subhiyakto and D. W. Utomo, “Software Testing Techniques and Strategies Use in Novice Software Teams,” *J. Sist. Inf.*, 2016.